

WHAT IS CLAIMED IS:

1. A method for emulating VGA hardware for a guest operating environment in a host operating environment for a computer system, wherein the guest operating environment is operable to send instructions to the VGA hardware, wherein the VGA hardware is configured under a current VGA mode and is operable to be configured in one of a possible plurality of VGA modes, and wherein the computer system comprises a memory, comprising the steps of:

receiving an instruction issued from the guest operating system and directed to the VGA hardware, wherein the instruction is associated with the current VGA mode;

branching to a jump table, wherein the jump table comprises one or more jump table entries operable for pointing to one or more addresses in memory and wherein the jump table entries are associated with a specific VGA mode;

pointing to an address in memory corresponding to a function, wherein the function is operable to allow the host operating system to execute the instruction; and

executing the function under the host operating environment.

2. A method for emulating VGA hardware for a guest operating environment in a host operating environment for a computer system, wherein the guest operating environment is operable to send instructions to the VGA hardware, wherein the VGA hardware is configured under a current VGA mode and is operable to be configured in one of a possible plurality of
5 VGA modes, and wherein the computer system comprises a memory, comprising the steps of:

receiving an instruction issued from the guest operating system and directed to the VGA hardware, wherein the instruction is associated with the current VGA mode;

branching to a jump table, wherein the jump table comprises one or more jump
10 table entries operable for pointing to one or more addresses in memory and wherein the jump table entries are associated with a specific VGA mode, and each jump table entry is associated with a specific instruction in the set of permissible instructions for the specific VGA mode;

pointing to an address in memory corresponding to a function generator, wherein
15 the function generator is operable to create a function, wherein the function is operable to allow the host operating system to execute the instruction;

storing the function in a selected location in the memory at a specific memory
20 address;

updating the jump table entry corresponding to the instruction with an entry
operable for pointing to the memory address corresponding to the function;

pointing to the function; and

25 executing the function under the host operating environment.

3. The method of claim 2, further comprising the steps of

providing a mode table, wherein the mode table comprises one or more mode
table entries, wherein each mode table entry is associated with a specific VGA mode and is
25 operable for pointing to one or more addresses in memory; and

updating the mode table entry corresponding to the current VGA mode with an
entry operable for pointing to the function, after the step of storing the function.

5. The method of claim 2, wherein the instruction is operable to change a pixel stored in the memory.

THE **OF** **AND** **TO** **FROM** **BY** **WITH** **IN** **ON** **AT**

6. A method for emulating VGA hardware for a guest operating environment in a host operating environment for a computer system, wherein the guest operating environment is operable to send instructions to the VGA hardware, wherein the VGA hardware is configured under a current VGA mode, and is operable to be configured in one of a possible plurality of VGA modes, wherein the current VGA mode is associated with a set of current mode flags, and wherein the computer system comprises a memory, comprising the steps of:

receiving an instruction issued from the guest operating system and directed to the VGA hardware, wherein the instruction is directed to change the current VGA mode to a new VGA mode;

generating a set of new VGA mode flags associated with the new VGA mode; and
comparing the current VGA mode flags to the new VGA mode flags.

7. The method of claim 6, further comprising the step of:

providing a jump table, wherein the jump table comprises one or more jump table entries operable for pointing to one or more addresses in memory, wherein the jump table entries are associated with a specific VGA mode, and each jump table entry is associated with a specific instruction in the set of permissible instructions for the specific VGA mode, and wherein the jump table contains the current VGA mode flags.

8. The method of claim 7, further comprising the steps of:

determining that there is no significant difference between the current VGA mode flags and the new VGA mode flags, after the step of comparing the current VGA mode flags to the new VGA mode flags; and

maintaining the jump table entries.

9. The method of claim 7, further comprising the step of:

providing a mode table, wherein the mode table comprises one or more mode table entries, wherein each mode table entry is associated with a specific VGA mode and is operable for pointing to one or more addresses in memory.

10. The method of claim 9, wherein each mode table entry is associated with a set of mode flags associated with the specific VGA mode.

11. The method of claim 10, further comprising the steps of:

5 determining that there is a significant difference between the current VGA mode flags and the new VGA mode flags, after the step of comparing the current VGA mode flags to the new VGA mode flags; and

 changing the jump table entries to entries that are operable to pointing to a memory address corresponding to a search function, wherein the search function is operable to compare the mode flags associated with each mode table entry to the new VGA mode flags.

12. The method of claim 10, further comprising the steps of:

 determining that there is a significant difference between the current VGA mode flags and the new VGA mode flags, after the step of comparing the current VGA mode flags to the new VGA mode flags; and

 comparing the new VGA mode flags to the mode flags associated with each mode table entry.

13. The method of claim 12, further comprising the steps of:

20 finding mode flags associated with a mode table entry that match the new VGA mode flags after the step of comparing the new VGA mode flags to the mode flags associated with each mode table entry; and

 copying the matching mode table entry to the jump table entries such that the jump table entries are operable to point to the same addresses in memory as the mode table entry.

25 14. The method of claim 13, further comprising the step of changing the current VGA mode flags contained in the jump table to the new VGA mode flags.

15. The method of claim 12, further comprising the steps of:

determining that none of the mode flags associated with the mode table entries match the new VGA mode flags;

setting of the jump table entries such that the jump table entries are each operable
5 to point to the address corresponding to a function generator, wherein the function generator is
operable to create a function, wherein the function is operable to allow the host operating system
to execute the instruction.

16. The method of claim 15, further comprising the step of changing the current VGA mode flags contained in the jump table to the new VGA mode flags.